

Amendment to the Claims:

This listing of claims will replace all prior versions, and listing of claims in the application:

Listing of the Claims:

1. (Canceled)
2. **(Currently Amended)** In a Java computing environment, a Java heap suitable for storing Java objects therein, wherein said Java heap comprises:
A Java computing environment as recited in claim 1,
at least one Java heap portion that is designated for storing Java objects with similar traits; and
wherein said at least one Java heap portion is designated to store objects of a class, and wherein only objects of said class are stored in said at least one Java heap portion.
3. **(Currently Amended)** A Java computing environment as recited in claim ~~[[1]]~~ **2**, wherein said Java heap includes a plurality of Java heap portions, and wherein each of said plurality of Java heap portions is designated for storing Java objects with similar traits.
4. (Original) A Java computing environment as recited in claim 3, wherein each of said plurality of Java heap portions is designated to store objects of the same size.
5. **(Currently Amended)** A Java computing environment as recited in claim ~~[[1]]~~ **2**, wherein said at least one Java heap portion is implemented as an array.
6. **(Currently Amended)** A Java computing environment as recited in claim ~~[[1]]~~ **2**, wherein said at least one Java heap portion is designated to store objects with similar life spans.

7. **(Currently Amended)** A Java computing environment as recited in claim [[1]] 2, wherein said at least one Java heap portion is designated to store objects of the same size.

8. **(Currently Amended)** A Java computing environment as recited in claim [[1]] 2, wherein said at least one Java heap portion is designated to store objects that do not reference other objects.

9. (Canceled)

10. (Original) A method for creating Java objects in a heap, said method comprising:
translating a Java Bytecode into one or more commands, said Java Bytecode representing an instruction for creating a Java object, and wherein said one or more commands can operate to allocate said Java object in a portion of heap designated for said object; and

executing said one or more commands to create said object in said portion of said heap associated with said object.

11. (Original) A method as recited in claim 10, wherein said method further comprises:

marking a Java Bytecode representing an instruction for creating a Java object.

12. (Original) A method as recited in claim 11,
wherein said marking is performed at compile time; and
wherein said one or more commands are created at compile time.

13. (Original) A method as recited in claim 10,
wherein said portion of the heap is designated for allocation of objects with similar traits; and
wherein objects that do not have similar traits as said object are not allocated in said portion of the heap.

14. (Original) A method of creating Java objects, said method comprising:
compiling one or more commands suitable for allocation of Java objects;

executing said one or more commands to allocate said Java objects in a designated portion of heap memory; and

wherein said Java objects are created in said designated portion of heap memory.

15. (Original) A method as recited in claim 14, wherein said Java objects have a similar trait.

16. (Original) A method as recited in claim 15, wherein Java objects that do not have said similar traits are not allocated in said designated portions of heap memory.

17. (Original) A method as recited in claim 16, wherein said similar traits can be: class, size, number of fields, or life span of said Java objects.

18. (Original) A computer readable medium including computer program code for creating Java objects in a heap, said computer readable medium comprising:

computer program code for translating a Java Bytecode into one or more commands, said Java Bytecode representing an instruction for creating a Java object, and wherein said one or more commands can operate to allocate said Java object in a portion of heap designated for said object; and

computer program code for executing said one or more commands to create said object in said portion of said heap associated with said object.

19. (Original) A computer readable medium as recited in claim 18, wherein said computer readable medium further comprises:

computer program code for marking a Java Bytecode representing an instruction for creating a Java object.

20. (Original) A computer readable medium as recited in claim 19,

wherein said marking is performed at compile time; and

wherein said one or more commands operating to allocate said Java object in said portion of said heap are created at compile time.

21. (New) A virtual machine for creating Java objects in a heap, wherein said virtual machine is capable of:

translating a Java Bytecode into one or more commands, said Java Bytecode representing an instruction for creating a Java object, and wherein said one or more commands can operate to allocate said Java object in a portion of heap designated for said object; and

executing said one or more commands to create said object in said portion of said heap associated with said object.

22. (New) A virtual machine as recited in claim 21,
wherein said one or more commands are created at compile time.

23. (New) A virtual machine as recited in claim 21,
wherein said portion of the heap is designated for allocation of objects with similar traits; and
wherein objects that do not have similar traits as said object are not allocated in said portion of the heap.

24. (New) A virtual machine as recited in claim 21, wherein said virtual machine is further capable of:
compiling one or more commands suitable for allocation of Java objects;
executing said one or more commands to allocate said Java objects in a designated portion of heap memory; and
wherein said Java objects are created in said designated portion of heap memory.

25. (New) A virtual machine as recited in claim 21, wherein said Java objects have a similar trait.

26. (New) A virtual machine as recited in claim 24, wherein said similar traits can be: class, size, number of fields, or life span of said Java objects.